

HTML CSS Notes

Tuesday, July 21, 2020

10:35 AM

SECTION 2: DIVE INTO HTML

6. What is HTML?

- Hyper Text Markup Language
- HTML documents are described by HTML tags
 - o `<h1>` header
 - o `<p>` paragraph
 - o `<a>` links
- Mark up beginning and end of an element:
 - o `<tagname>content</tagname>`

7. Structure of an HTML Document

- Index.html is the standard name of the main HTML document
- Head and body is the main elements of an HTML doc
 - o Head holds the title, css style, and other information for the browser
 - o Body holds the visible contents
 - o `<!DOCTYPE html>` must be the first thing in an html file which lets the browser know the type of document type
 - o [add extension autosave in brackets]

8. Starting to fill the structure

- Body can include headings starting with h1 for important headings all the way down to h6 for less important headings
- Some tags
 - o `<title>` belongs in the head element. Creates title for the webpage
 - o `<h1>` `<h2>` `<h3>` Creates headings in descending importance/size
 - o `<p>` paragraph for longer text
 - o `` bolds text

- `` italicizes text
- `<u>` underlines text
- `
` creates breaks in paragraphs

9. Images and Attributes

- `` tag does not need a closing tag
- Attributes of ``
 - Src - image file path
 - Alt - in case the image cannot be displayed for some reason an alternative text describing the image will be displayed instead

10. Links

- Links are defined with `<a>` tag which stands for anchor
- Attributes
 - Href - link to website or photo
 - Target="_blank" - allows to open a new tab once link is clicked

SECTION 3: FORMATTING WITH CSS

11. Getting Started with CSS

- Cascading Style Sheet
- CSS defines how HTML should look
- HTML is content
- CSS is style
- There are three ways to use CSS
 1. Inside HTML tag using the style attribute
 2. Inside HTML document using style tag
 3. External .css file separate from the HTML file (best)
- Need to link the html file with the css file using the following line inside the head element of the HTML file: `<link rel="stylesheet" type="text/css" href="style.css">`

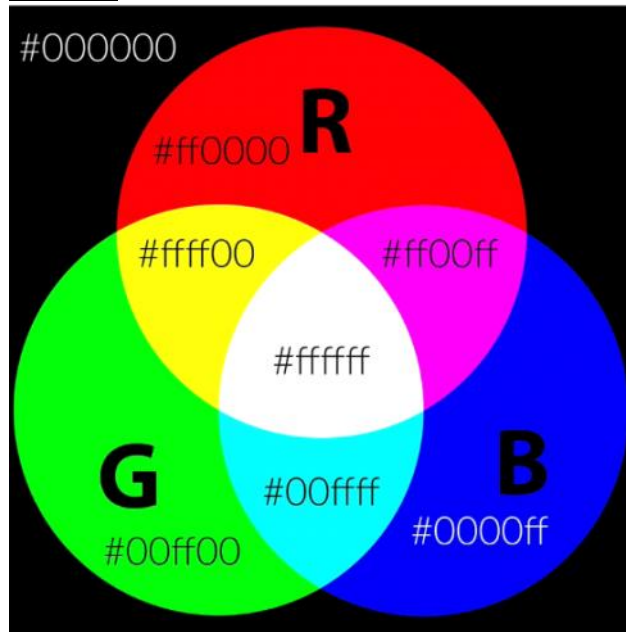
12. Starting to Make Our Webpage Pretty: text

- CSS is written in rules. Each rule consists of a selector and a

declaration block

- Selector selects which elements from the html doc to style
- Declaration is enclosed in {} and consists of all the styling
- CSS implements inheritance.

13. Colors



#RRGGBB

For each color:

Min: 0 (hex 0)

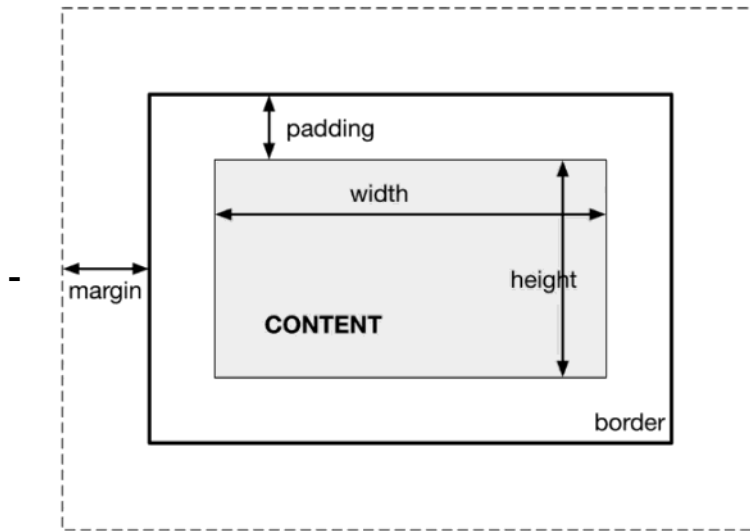
Max: 255 (hex ff)

- Transparency uses the rgba notation

14. Classes and ID's

- We can attribute class or ID names to html elements for a more specific targeted styling
- Classes can be used multiple times
- ID can only be used once inside a class
- In the CSS file, use the . Operator to select classes and # to select ID
 - .main-text
 - #author-text

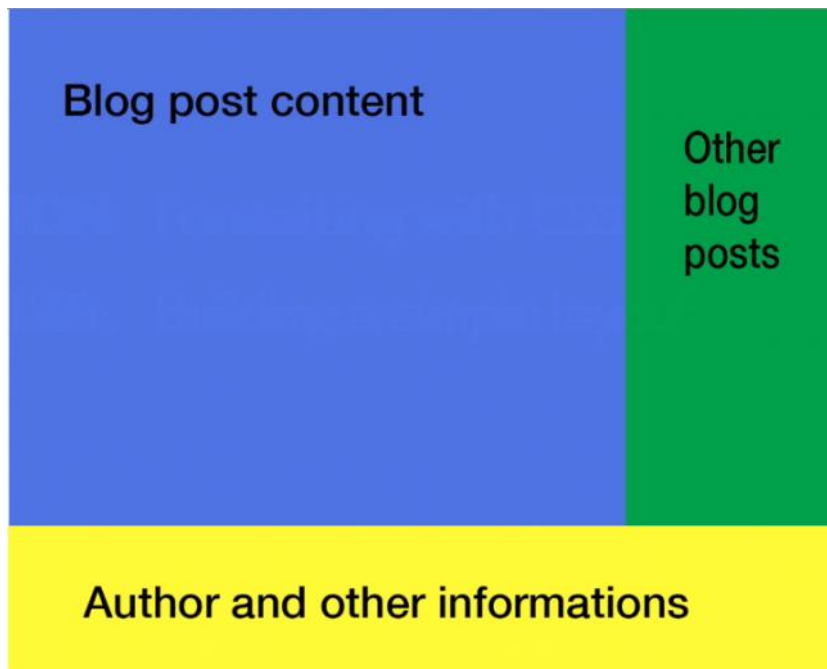
15. The CSS box Model



- **Content:** text, images. etc.
- **Padding:** transparent area around the content, inside the box.
- **Border:** goes around the padding and the content
- **Margin:** space between boxes.

- Padding margin border are attributes in css
- Width and height in css determines the dimensions of the content and not the box itself. Use css property call box sizing and set to border box to include padding and border when sizing using height and width
- Block and inline elements
 - Block elements (headings, paragraphs) use the full width of the browser and forces line breaks
 - Inline elements (links, images) does not do that. Height and width cannot be set
- Use the * operator to select all elements in the html doc

16. Building a Simple Layout



- Div element - divide page into sections
- Float property - an element can be pushed to the left or right allowing other elements to wrap around it
- Margin: [top] [right] [bottom] [left] short hand
- Border: [thickness] [style] [color]

17. Polishing our Blog Post

- .class img{} selects the image in the class without having to create another class for that image

18. Relative vs Absolute

- So far our elements have been positioned relative to other elements

SECTION 4: WEB DESIGN BASICS

21. Typography

- Size
 - Body text: 15-25px
 - Headline: bigger sizes (no real range or limit)
 - The bigger the lesser the weight so it does not steal most of attention

- Line spacing: 120 - 150% for easy read
 - Char length per line: 45 - 90
 - Fonts: sans-serif (clean, modern, simple) serif (traditional, long reading)
- How to choose a font?
 - Reflect the look and feel of the website
 - Decide a type face (sans-serif or serif)
 - Be consistent

22. Colors

- Use single base color
 - flatUI colors (resource)
- Create a color palette from the base color
- Use color to draw attention
- Red: passion, strength
- Orange: draws attention, cheerfulness, friendliness, courage, creativity
- Yellow: energetic, happiness, curiosity, brightness
- Green: harmony, life, health, money
- Blue: patience, peace, trust, stability, professional
- Purple: power, wealth, wisdom, royalty
- Pink: romance, affection, peace
- Brown: relaxation, earthy, natural, durability, comfort

23. Working with Images

- Overlay image with a color or color gradients to contrast text over the image
- Put text in opaque color boxes to make text stand out
- Overlay text on blurred images
- Floor fade: image fades to black near the bottom allowing white text to stand out
- Where to get amazing images
 - Google

24. Working with Icons

- Icons should not take center stage
- Use Icons to list features/steps
- Use icons for actions and links
- Icons should be easily recognizable
- Label your icons
- Use vector icon font (smoother)

25. Spacing and Layout

- Put white space between
 - o Elements
 - o Group of elements
 - o Website sections
- Do not exaggerate
- Define where you want audience to look first
- Establish a flow that corresponds to your content's message
- Use whitespace to build that flow

27. Inspiration

- Look at other websites
 - o Why do they look good?
 - o What do these sites have in common
 - o How were they built in HTML and CSS
- Steal like an artist
 - o Don't worry too much about being original, but do not copy literally

SECTION 5: THE KILLER WEBSITE PROJECT

32. The 7 Real World Steps to a Fully Functional Website

1. Define your project

- Start off by defining the goal of your project. This can be showing your portfolio to the world, selling an e-book, building a blog, etc.
- Also define your audience. Ask yourself: which is the typical user

that will visit my website?

- This is important, because you should always design with your goal and audience in mind.

2. Plan out everything

- Once your project is defined, plan your content carefully. This includes text, images, videos, icons, etc.
- Remember what I told you about visual hierarchy. It plays an important role when you start thinking about what you want on your website and what you don't.
- Defining the content before actually starting the design is called the content-first approach. It means that you should design for the content, instead of designing a webpage and then filling it with some stuff.
- Define the navigation.
- Define the site structure. You can draw a sitemap in this step if we're talking about a bigger project.

3. Sketch your ideas before you design

- Now it's time to get inspired and think about your design.
- Then, get the ideas out of your head. And with that I mean that you should sketch your ideas before you start designing. It will help you explore ideas and create a concept of what you want to build. Using pencil and paper is a great way of quickly retaining your valuable ideas.
- Make as many sketches as you want, but don't spend too much time perfecting anything. Once you have an initial idea, you can concentrate on the details when designing in HTML and CSS.
- I advise you to never, ever start designing without having an idea of what you want to build. Getting inspiration is very important in this phase, and I already told you how to do that!

4. Design and develop your website

- After sketching, start to design your website using all the guidelines and tips you've learned in the web design section.
- You'll do that using HTML and CSS, which is called designing in the

browser. Designing in the browser is basically designing and developing at the same time.

- There are more and more designers leaving traditional design programs such as Photoshop and start designing in the browser. The biggest reason for this is that you can't design responsive websites in photoshop. It also saves you tons of time.
- In this phase, you'll use your sketches, content and planning decisions you've made in steps 1, 2 and 3.

5. It's not done yet: optimization

- Before you can actually launch your beautiful masterpiece for the world to see it, we have to optimize its performance in terms of site speed.
- You also need to do some basic search engine optimization (SEO) for search engines such as google.

6. Launch the masterpiece

- Your optimized website is now finally ready to launch.
- All you need for launching is a webserver that will host your website and deliver it to the world.

7. Site maintenance

- The launch of your website is not the end of the story.
- Now it's time to monitor your users' behavior and make some changes to your website if necessary.
- You should also update your content regularly in order to show your users that your website is alive! For instance, a blog can be a great way of doing that.

36. First Development Steps

- Download normalize.css
- Folder creation
 - Resources (our stuff)
 - Css
 - img

- Js
 - Img
 - Data
- Vendor (stuff we got from the internet/other people)
 - Css
 - Js
 - Fonts
- Start with this setup in the css file


```

* {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
}
html {
  background-color: #fff;
  color: #555;
  font-family: 'Lato', 'Arial', sans-serif;
  font-size: 20px;
  font-weight: 300;
  text-rendering: optimizeLegibility;
}
```

37. Setting up Fluid Grid for Responsive Web Design

- Responsive design meaning that the website will work in any resolution, screen size, browser, phone or laptop.
- 50% of internet traffic comes from mobile phones. Mobile friendly websites will be ranked higher in google search

Three ingredients to responsive webdesign

- **Fluid Grid:** all elements are sized in relative units (% instead of px)
- **Flexible images:** also sized in relative units
- **Media queries:** allow us to specify different CSS style rules for different browser widths
- Download responsivegridsystem.com
- Short hand from margin: 0 auto 0 auto is margin: 0 auto

38. Building the Header - Part 1

- Html elements: header, nav, ul, li
- Put text on image: make image darker
- How to make that image as high as the browser viewport
- How to make a vertically a horizontally centered box
- How to design buttons
- The 4 link states in CSS: link, visited, hover, active
- CSS3 transitions for small animations
- How to create a simple navigation
- Header element is similar to a div, but it's a more specific container that tells the browser that this is our header
- Setting background image will only fill the box. Want the image to fill entire viewport
 - o Height: 100vh (100% of the view port)
- Inspect each element to see what is contributing to the white space
- Centering the text
 - o Set text box hero element to be the same max width as its container
 - o Set the text to have an absolute position
 - o Move the top left corner point of the text box at the 50% mark from the top and 50% mark from the left of its parent container (row class element)
 - o Translate the text box itself to the move 50% to the right and 50% down of its width and height respectively
- To make background image darker use linear gradient from black to black with some transparency

39. Building the Header - Part 2

- Font-size percentage is relative to the global font size set in the html element
- Class="btn btn=full" btn is the parent class of btn-full
- To give an inline element some block properties (padding, margins, etc) use the property display: inline-block;
- Pseudo classes, states of elements (visited, clicked hovered)

- Transitions from different states in the elements

41. Building the Features - Part 1

- Learn
 - How to put content on a website directly from the CSS with the :after pseudo class
 - Use of fluid grid
 - Use of icons
- HTML special characters (ie. —)
- One way to center a block is to set width to a percentage (ie. 70%) and margin-left to be $(100\% - \text{width\%})/2$

43. Building the Favorite Meal Section

- Figure is another HTML elements that can contain images, and a caption for an image
- Technique for making images darker
 - Set an opacity
 - Set background color to black

45. Building the how-it-works Section

- Drawing circles with css
- Technique to make separations between sections by changing the background color slightly
- How to incorporate App Store and play Store buttons
 - Instead an <a> element, instead of a text content, we can put an image
- Selecting a box individually
 - Psuedo-class :first-child
 - Psuedo-class :last-child
- Text-align property works for any inline element
- Clearfix class applied to the parent element of the element that has a float property

```
.clearfix {zoom: 1}
.clearfix: after {
  content: '.';
```

```
clear: both;
display: block;
height: 0;
visibility: hidden;
}
```

47. Building the city section

- Practice new skill set
- How to use small icons aligned with some text
- How to style generic links

49. Building Customer Testimonials

- Html element for quotes: <blockquote>
- HTML element for citing: <cite>
 - An image can be included inside the cite element
- *When we make an element to have absolute positioning, we have to make its parent relative positioned.
- :before pseudo class
- Putting content in the document through CSS using content property

51. Building the Sign up Section

- Design with border radius
- Create box shadows with CSS
- to style text separately from the rest of the paragraph

52. Building Contact Form

- Use a set of new HTML elements to create forms, label, input, select, option, text area
- And style these elements

55. Building the Footer

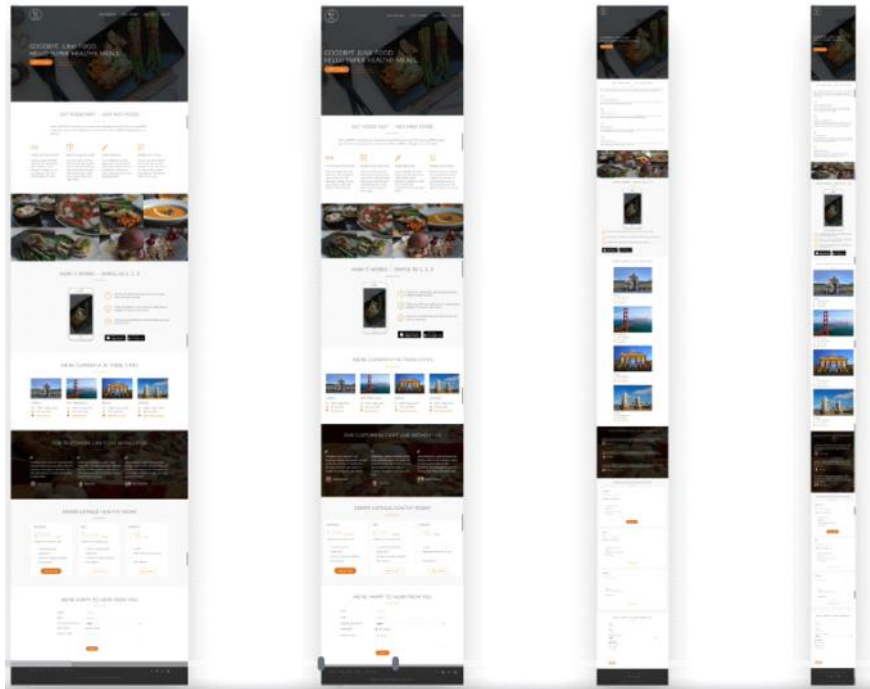
- How to build a simple but effective footer

- A cool way of using social icons

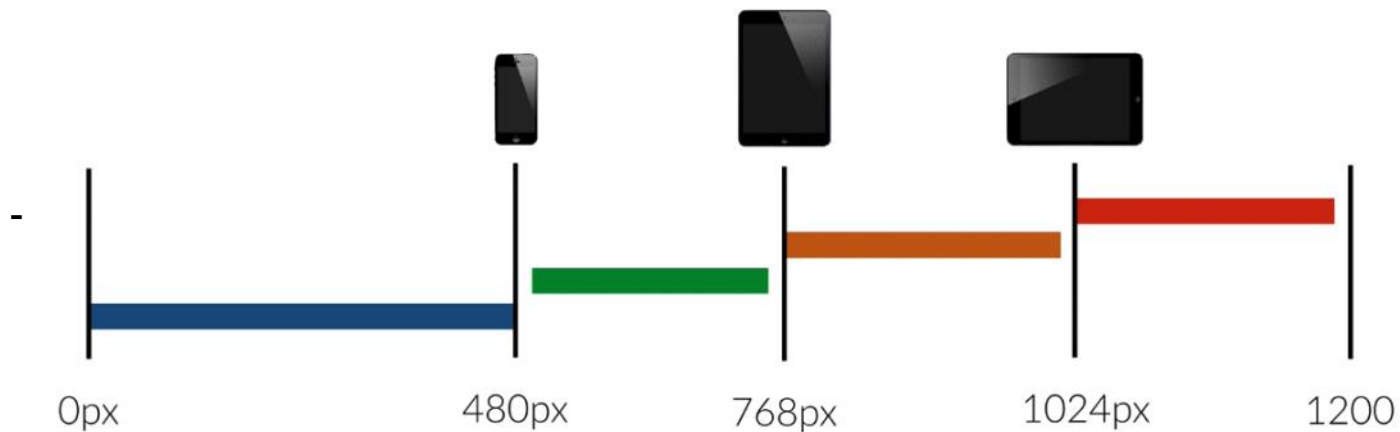
SECTION 6: RESPONSIVE WEB DESIGN WITH MEDIA QUERIES

57. Making the website responsive

- Want to make website shine in all screen sizes



- Media queries and break points



- Media queries are what enables the browser to call css declaration based on the width of the screen. These queries will trigger at different break points

- Break points are screen width at which we want our queries to activate

59. A Note About Web Browsers

- Majority of users use Chrome
- Css browser Prefixes (find extensions that does the prefixing for you)
 - Android: -webkit-
 - Chrome: -webkit-
 - Firefox: -moz-
 - IE: -ms-
 - iOS: -webkit-
 - Opera: -o-
 - Safari: -webkit-
- Jsdeliver.com for scripts
 - Respond
 - Html5shiv
 - Selectivizr
- Caniuse.com to see what exactly works in which browser

SECTION 7: LET'S ADD SOME COOL EFFECTS

61. Introduction to jQuery

- jQuery is the most popular library in the world
- Free
- jQuery makes it easy to
 - Select and manipulate HTML elements
 - Create animations
 - Develop Ajax Applications
- Developers are able to make plug ins
- Magnific pop up jQuery plug in
 - Pop ups
- Tooltipsters plug in

- Cool hover tips
- Maplace.js plug in
 - Helps include google maps
- Typer.js plugin
 - Types
- Onepage scroll plug in
 - Scrolls one full page at a time
- To include jQuery, use google hosted library distribution network
- Every document we put our js will need the following line
 - `$(document).ready(function() {`
 / insert code here
 /insert code here
 - `})`
- This will cause our code will run only after the page has been loaded

62. Building a 'sticky' navigation

- A way to keep our navigation on screen as we scroll down
- We are going to create a sticky class to include in our index
- First we format
 - Keep position fixed
 - Play with display options: none, block
- Download waypoints js plug in
- Convention: to leave our js script as the last script on the html document
- Recommended to use another class in HTML elements for js selector.
 - `le <section class="section-features js--section-features">`
 <section>
- z-index css property for element stacking on a webpage

64. Scrolling to Elements

- .animate method

65. Animations on scroll

- Css plugin: animate.css

66. Responsive Navigation

- Hamburger icon as a button to open vertical listed navigation bar

68. Enhancing page speed

- Optimize heavy images
- Minify css and jquery code
- Want size to be twice larger for resolution purposes
- Use optimozilla to reduce image sizes
- Use css compressor (minifycss.com)

69. Search Engine Optimization

- Techniques that improve and promote the number of a webpage visit
- Meta description tag
 - o No more than 160 characters recommended

```
<meta name="description" content="Udemy is the world's largest destination for online courses. Discover an online course on Udemy.com and start learning a new skill today."/>
```



Udemy: Online Courses - Anytime, Anywhere

<https://www.udemy.com/> ▾ Traduzir esta página

Udemy is the world's largest destination for online courses. Discover an online course on Udemy.com and start learning a new skill today.

- HTML Validation
 - Validatorw3.com
 - Following the official HTML rules and has no bugs.
 - Valid HTML is preferred by google. Better for accessibility.
- Content is king
 - Have great content in your website

- Keep providing new content if you can
- Keywords
 - Do not spam keywords - can be penalized
 - Use keywords, in title, metatags, headings and links
- Backlinks
 - Have other websites to link to your site

70. Launch website to the internet

1. Choose and buy a domain name (www.omnifood.com)
2. Buy web hosting: folder in the internet where we put our files in order to ensure our website runs smoothly, we need the right amount of bandwidth, and that is the amount of data transfer within a time frame
3. Upload our website

(ie. Godaddy website)

Attach /cpanel at the end of the domain name to manage website (if created from godaddy.com)

71. Google Analytics

- keep track of websites behaviour

72. What now?

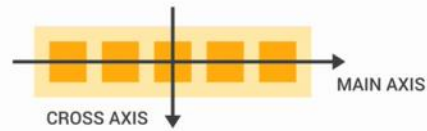
- Redesign websites you think are poorly designed
- Design a website for free for charity
- Joining design contests at 99design.com
- Freelance work freelancer.com, odesk, elance

SECTION 8: BONUS LECTURE

74. What's new in CSS

- Flexbox and css grids

FLEXBOX PROPERTIES OVERVIEW



CONTAINER

- 1 **flex-direction:** row | row-reverse | column | column-reverse
- 2 **flex-wrap:** nowrap | wrap | wrap-reverse
- 3 **justify-content:** flex-start | flex-end | center | space-between | space-around | space-evenly
- 4 **align-items:** stretch | flex-start | flex-end | center | baseline
- 5 **align-content:** stretch | flex-start | flex-end | center | space-between | space-around

ITEM

- 1 **align-self:** auto | stretch | flex-start | flex-end | center | baseline
 - 2 **order:** 0 | <integer>
 - 3 **flex-grow:** 0 | <integer>
 - 4 **flex-shrink:** 1 | <integer>
 - 5 **flex-basis:** auto | <length>
- flex:** 0 1 auto | <int> <int> <len>